## Status and todo

This is the status as of Thu May 4 22:11:42 CEST 2017.

### What has been done

1. Fastnetmon notification script uploads *rule files* correctly to the database server.
2. `db2dps` on the database server reads the rule files and inserts an aggregated version in the database. (see db2dps documentation)
3. `db2dps` reads the database for new rules and correctly converts them to flowspec rules, which are sent to exabgp. Only destination addresses within forskningsnet are processed and the package length may be given in the database as either
    1. `number`
    2. `list of numbers` separated by white space only
    3. `<number`, `>number` (greater and less than number)
    4. `number-number` (range between numbers) (as well as `=number` and `=number =number ...` meaning (a list of) port numbers).
    Combinations are *not allowed* as exabgp cannot parse it.
4. `source`, `destination` and `source/destination` port may be specified the same way and with the same limitations. IP address which is not single addresses must be in CIDR format. If the source or destination address is a single host, the address is converted to `a.b.c.d/32`. Only IPv4 addresses are accepted.

### Order of priority for the next step

1. Fix problem where `db2dps` seems to sleep the initial *sleep time* and not recognizing first round.
2. Find annoying error with sending flowspec rules to exabgp2; it works for exabgp1. It is not due to ssh configuration errors …
3. Attacks based on `ip_fragmentation_flood`, `DNS amplification`, `NTP amplification`, `SSDP amplification` and `SNMP amplification` needs further investigation (I need to see what fastnetmon prints upon detection).
4. Everything labelled `TODO:` (counting to 4).
5. The kill switch must write information back to the database, talk with Kasper Sort about it: it *must not be a hack*, and *should not be* interpreted by the rule generator.
6. Build a `fnm2db-conf.pl` which will
    - `$0 -a addres`: fetch and add the public key from
    - `/opt/i2dps/etc/ssh/id_ed25519.*` to `/home/sftpgroup/newrules/.ssh/authorized_keys` while deleting

existing / redundant keys for *address*.

- `$0 -u address` : push / build all configuration files for *address*: `/etc/fastnetmon.conf` , `/etc/networks_list` and `/etc/networks_whitelist` : fastnetmon version specific configuration files `/opt/i2dps/etc/fnm2db.ini` : configuration file for `fnm2db.pl` , the fastnetmon notification script

7. Write / update / change documentation
8. Code review on `db2dps` preferable with FTH
9. Install OpenVPN / pFsense with a Linux / Debian client configuration. Decide if the security layer should be on the ssh keys or the OpenVPN configuration - or both.

Incorporate the following in the documentation (sftp subsystem lockdown):$

- limiting access with sftp jail
- openssh restrict to sftp chroot
- restrict sftp user home

Notice the beauty of `root` as owner of `~/` .

Notice that *if the password field* in `/etc/shadow` is `:!:` then the user *cannot login with ssh*, change it to `:*:` to enable ssh login while preventing the user from changing password; as the magic string is not a valid result of crypt(3). Notice that this is not security by obscurity , its just old fashioned obscurity.